

Métricas para Métodos ágeis de Desenvolvimento – um Estudo Comparativo

Angelica Toffano Seidel Calazans
angelica.calazans@uniceub.br
Uniceub

Marcia Silva de Alvarenga
hemacia@hotmail.com
Uniceub

Resumo: O objetivo geral foi identificar e analisar as métricas voltadas para os métodos ágeis e contextualizar a sua atuação. Para isso foram conceituados o processo de desenvolvimento ágil, identificados os métodos ágeis mais utilizados e suas etapas. Identificaram-se as métricas de tamanho, esforço, custo e qualidade e, foi analisada a aplicabilidade das métricas mais citadas. O método adotado foi o estudo de caso e os instrumentos de coleta de dados foram: a revisão de literatura e o método Delphi com Q-sort. O instrumento elaborado para aplicação do método Delphi com Q-sort foi construído considerando a revisão de literatura elaborada e foram classificadas métricas referentes a produtividade e qualidade. O estudo de caso foi realizado na área de desenvolvimento de software do Uniceub, onde trabalham 22 indivíduos. Foram aplicadas 3 rodadas do Delphi com Q-sort e a terceira rodada apresentou um nível de concordância satisfatório. Para análise dos dados utilizou-se o software estatístico SPSS, o coeficiente Kendall's W e coeficiente de correlação rho de Spearman. Os resultados mostram que, no contexto do estudo de caso analisado, as métricas Estórias (ou funcionalidades) entregues, Lista de atividades concluídas e estimativas finais foram as métricas de produtividade consideradas de maior aplicabilidade. Foram consideradas as métricas de produtividade com menor aplicabilidade: pontos de função entregues, linhas alteradas e linhas de código. Com relação as métricas de qualidade foram identificadas com maior aplicabilidade: Cliente – satisfação do produto, Defeitos de resolução e Equipe-moral. E como métricas de menor aplicabilidade: Fator de teste, Commits e Configuração.

Palavras Chave: métricas - métodos ágeis - Delphi com Q-sort - produtividade - qualidade

1. INTRODUÇÃO

A construção de um produto de software é uma tarefa complexa. Existem várias metodologias para o desenvolvimento de um produto de software, entre elas as chamadas metodologias ágeis ou métodos ágeis. O foco principal dos métodos ágeis de desenvolvimento de software é a simplicidade e a velocidade (VICENTE, 2010). A equipe de desenvolvimento ágil concentra-se na entrega rápida de funcionalidades realmente necessárias aos seus usuários, coletando suas opiniões (feedbacks) e procurando reagir rapidamente às mudanças negociais e/ou tecnológicas ao longo do projeto de software.

O manifesto de desenvolvimento ágil de software é apontado como sendo o guia para todos os métodos de desenvolvimento considerados ágeis (VICENTE, 2010). Esse manifesto foi lançado em 2001 com o intuito de discutir ideias e procurar alternativas aos processos burocráticos e às práticas adotadas nas abordagens tradicionais de engenharia de software e gerência de projetos. O manifesto define os seguintes valores a serem utilizados por todos os métodos ágeis (WILLIAMS, 2007; KNIBERG, 2007):

- Indivíduos e interações são mais importantes que processos e ferramentas;
- Software funcionando é mais importante que documentação completa e detalhada;
- Colaboração com o cliente é mais importante que negociação de contratos;
- Adaptação às mudanças é mais importante que seguir um plano.

Mas, independentemente da metodologia adotada para o desenvolvimento do produto de software a dinâmica de mercado atual tem imposto às organizações a necessidade de terceirizar a construção de seus sistemas, para dessa maneira conseguir atuar no mercado de forma mais rápida e eficiente. Para implementar essa terceirização, as leis e regulamentos brasileiros tais como IN04 (BRASIL, 2008), Lei 8666/93 (BRASIL, 1993) citam ser necessárias a utilização de métricas para a mensuração do tamanho e estimativas de esforço, prazos, custos e qualidade.

No contexto de medições, constantemente surgem novas métricas voltadas para estimativas especificamente utilizadas em métodos ágeis. As métricas mais conhecidas no contexto de métodos ágeis são Story Points e Planning Poker, mas encontram-se trabalhos acadêmicos que focam fatores de esforço, complexidade e qualidade para métodos ágeis tais como Ram (2009), Bhalerao & Ingle (2009) etc.

Considerando essas questões pode-se levantar os seguintes questionamentos, os quais serão respondidos por esta pesquisa:

- Quais os métodos ágeis mais citados pela literatura?
- Quais as métricas de tamanho, esforço, custo e qualidade propostas para esses métodos?
- Quais as métricas mais aplicáveis ou adequadas para uma organização?

Assim, o objetivo geral desta pesquisa é identificar e analisar as métricas voltadas para os métodos ágeis e contextualizar sua atuação no processo ágil de desenvolvimento. São objetivos específicos deste trabalho:

- Identificar os métodos ágeis mais citados pela literatura, suas fases e atividades;

- Identificar as métricas propostas de tamanho, esforço e custo para esses métodos;
- Analisar a aplicabilidade das métricas mais citadas no contexto de um desenvolvimento ágil.

A seguir, a seção 2 apresentam-se alguns métodos ágeis. A seção 3 descreve medição e cita as métricas utilizadas para métodos ágeis e a seção 4 apresenta a metodologia da pesquisa. A seção 5 demonstra a análise dos resultados e, na seção 6 as conclusões obtidas.

2. METODOLOGIA ÁGIL

Para a criação de um produto de software é necessário a interação dos vários fatores, como processos e pessoas. A dinâmica de mercado atual impõe agilidade na entrega de produtos e atendimento, cada vez mais satisfatório, ao que foi demandado pelo cliente. Os métodos ágeis representam uma novidade diante das metodologias tradicionais porque valorizam a interação entre pessoas, a comunicação com o cliente e priorizam as atividades que efetivamente agregam valor. Encontram-se na literatura várias propostas de métodos ágeis de desenvolvimento de sistemas, alguns descritos sucintamente, a seguir.

2.1 EXTREME PROGRAMMING

O propósito da Programação Extrema (XP) é garantir o sucesso no desenvolvimento de softwares, especialmente para equipes pequenas e médias envolvidas com projeto, nos quais os requisitos são desconhecidos e voláteis. O XP foi formalizado por meio de quatro princípios que são: Comunicação, Simplicidade, Feedback e a Coragem. É composto de doze práticas (TANIGUCHI & CORREA, 2009; FADEL & SILVEIRA, 2010): planejamento das iterações, incrementos curtos e pequenos, uso da metáfora para facilitar a comunicação, projeto simples ou incremental, desenvolvimento orientado a testes, reestruturação constante do código, programação em pares, propriedade coletiva, integração contínua, ritmo sustentável, cliente no local e padrão de codificação.

Segundo FADEL & SILVEIRA (2010), o ciclo de desenvolvimento da Programação Extrema é composto de seis fases: exploração, planejamento, iterações para as versões, produção, manutenção e morte. A fase de exploração, engloba a descrição das histórias pelos clientes (user stories), das características e funcionalidades do software. Na fase de planejamento, os programadores calculam em semanas a estimativa de esforço para cada história e os clientes adaptam as histórias para esforço de no máximo três semanas. Na fase de iterações para as versões os desenvolvedores e o cliente escolhem histórias para cada iteração e o cliente pode determinar testes funcionais para serem aplicados ao final de cada iteração.

Ao final da última iteração, o sistema passa para a fase de produção, na qual são realizados testes de desempenho e outros testes necessários para a entrega do produto. Durante essa fase, também, é discutido sobre as solicitações de mudança e em qual versão serão dispostas. A fase de manutenção consiste no acréscimo ou manutenção das histórias no sistema em operação e quando não há mais histórias a serem executadas inicia-se a fase de morte. Vicente (2010) ressaltou que os times de desenvolvimento realizam quase todas as atividades de forma simultânea.

2.2 SCRUM

O SCRUM é uma metodologia ágil que visa fornecer software de forma rápida e com maior qualidade. Ele baseia-se em algumas características (SCHWABER, 1997): flexibilidade dos resultados e dos prazos, times pequenos, revisões frequentes e colaboração.

O SCRUM se concentra em descrever como os membros da equipe devem trabalhar para produzir um sistema flexível, num ambiente de mudanças constantes. A equipe SCRUM é formada por três principais papéis: *product owner* (PO), *SCRUM master* e *time SCRUM*. O PO tem a responsabilidade de priorizar e validar os requisitos. O SCRUM *máster* é responsável pelo macro gerenciamento e por manter o processo de SCRUM funcionando. O *time SCRUM* é responsável pelo micro gerenciamento do projeto. Segundo Fadel & Silveira (2010), o SCRUM é uma metodologia destinada a equipes com menos de dez pessoas.

Outra característica da metodologia SCRUM, segundo Carvalho & Mello (2012) é a existência de três principais artefatos produzidos e acessíveis por toda a equipe:

- Product Backlog - uma lista, ordenada por prioridades, com todas as necessidades e funcionalidades a serem desenvolvidas, os custos estimados e as datas de entrega.
- Sprint Backlog - é um subconjunto do Product Backlog, o qual divide as atividades por períodos de desenvolvimento chamados Sprint's. Geralmente, cada Sprint dura de uma a quatro semanas.
- Gráfico Sprint Burndown - é uma representação gráfica do trabalho já realizado e o que resta a ser realizado.

A metodologia SCRUM tem as seguintes fases: reunião de planejamento (seleção do Backlog e definição das estimativas), planejamento da sprint (backlog do sprint com atividades ou tarefas), sprint de desenvolvimento (análise, projeto, evolução teste, entrega) e reunião de revisão do sprint e retrospectiva.

2.3 OUTROS MÉTODOS ÁGEIS

A literatura apresenta outros métodos, tais como: Crystal, ASD (Desenvolvimento ágil de software), DSDM (Metodologia de Desenvolvimento de Sistemas Dinâmicos), FDD (Desenvolvimento Guiado por Funcionalidades) etc. Segundo Vicente (2010) esses métodos apresentam algumas variações (por exemplo: foco pequenos projetos ou médios e grandes projetos, etc.) mas respeitam os valores citados no manifesto de desenvolvimento ágil de software.

3. MEDIÇÃO E MÉTRICAS

Medição aplicada à área de Engenharia de Software é conceituada como a avaliação quantitativa de algum aspecto da engenharia de software, processo ou produto e tem como objetivos facilitar a análise, a estimativa e o controle do processo de desenvolvimento de um produto de software e estabelecer baselines para ajudar o desenvolvimento futuro (FENTON & PFLEEGER, 1997). Fenton & Pfleeger (1997) identificam 3 principais classificações de medições voltadas para o desenvolvimento de software: medição de processos, medições de produto e medições de recursos.

Sato (2008) e Cohn (2008) citam uma série de critérios que uma boa métrica ágil deve ter, tais como: reforçar princípios ágeis, envolvendo toda a equipe, seguir tendências e não números; pertencer a um conjunto pequeno de métricas e diagnósticos; ser facilmente coletada, deixar claro os fatores que a influenciam para evitar manipulações, facilitar a melhoria do processo, fornecer feedback etc.

Considerando essas premissas foram identificadas, a partir de 2007, por meio de revisão de literatura, algumas proposições de métricas e adaptações de métricas existentes para medir tamanho do produto, custo, prazo, recursos de projetos ágeis. A maior parte dos trabalhos encontrados sugerem propostas, embora poucos as apliquem em estudos de caso

acadêmicos ou da indústria. Foram também encontradas métricas para mensurar a qualidade de processo de projetos ágeis, para permitir o acompanhamento etc. É interessante ressaltar que, essas métricas poderiam ser utilizadas para avaliar, em caso de contratação externa, a qualidade do desse serviço.

Assim, a seguir são sucintamente descritas algumas das métricas classificadas em dois grupos: métricas de produtividade e de qualidade. As métricas relacionadas a produtividade tem como foco identificar tamanho, prazo, recursos para a construção do produto. As métricas de qualidade englobam avaliar a qualidade do processo, do produto de software e dos recursos utilizados. A Tabela 1 apresenta também se houve a aplicação prática das métricas.

Tabela 1: Relação de métricas propostas para métodos ágeis classificadas por Produtividade e Qualidade

<i>Publicação encontrada</i>	<i>Comentários / Aplicação Prática Existente</i>
<i>Métricas para produtividade (tamanho, esforço e custo)</i>	
Silva et al (2012) IFPUG (2010)	O tamanho do produto é mensurado pela APF – Análise de pontos de função. Essa métrica pontua as funcionalidades de um software. Classifica as funcionalidades em: funções de dados e de transações. Silva et al (2012) aplicaram a proposta em um sistema. Alguns editais do governo brasileiro estão contratando o desenvolvimento ágil com APF ¹ e citando IFPUG(2010)
Cagley (2009)	O tamanho do produto é mensurado pela Quick e Earl Function Points- QEFP (adaptação da APF) em combinação com o Delphi ² e Planning Poker ³ . Sugere uma fórmula para produtividade. CAGLEY (2009) aplicou sua proposta em dois estudos de caso com resultados bons.
Kang et al (2010)	Criou uma abordagem de modelo de estimativa de custo e monitoramento para projetos ágeis, utilizando a APF, Story points ⁴ e o filtro de Kalman ⁵ . Segundo os autores tanto a APF como o Story points podem ser utilizados com o filtro de Kalman. O autor aplicou em um projeto de ficção para demonstrar a abordagem
Gamba & Barbosa (2009)	Utilizam três métricas: Ideal Day ⁶ , Planning poker e APF em um estudo de caso (com 34 requisitos) em uma organização. Segundo os autores os resultados demonstraram que o Ideal Day e o Planning poker estimaram o tempo com maior precisão que a APF. Os autores aplicaram em um projeto com 34 requisitos
Ram (2009)	Autor propõe métricas de processo, produto e projeto focadas no desenvolvimento ágil para mensurar tamanho e esforço. Ex. Fator de complexidade do sprint, fator de esforço do sprint etc. RAM (2009) aplicou as métricas em duas dúzias de projetos SCRUM, variados com participação de 5 a 7 membros e 10 a 15 membros.
Bhalerao & Ingle (2009)	Propõe um método algoritmo para estimar tamanho, duração e custo de projetos ágeis considerando uma adaptação do story points para acrescentar

¹ AGU - Advocacia-Geral da União Secretaria-Geral Divisão de Compras (2011); SESI, Edital concorrência n.º 283/2009, (2009); TST (2012); STF(2012)

² Técnica que possibilita a obtenção de consenso por meio de sucessivos questionamentos a especialistas

³ Técnica que possibilita a obtenção de consenso distribuindo cartas com valores para especialistas opinarem

⁴ Medida de tamanho de uma história (story points) – pontos de história

⁵ Algoritmo de rastreamento que considera bases históricas e acompanhamento diário para estimar valor futuro

⁶ Técnica que considera o quantitativo de dias ideais para o desenvolvimento. Dias ideais correspondem a quantidade de horas utilizadas somente para o desenvolvimento (sem as interrupções, sem considerar as reuniões, etc)

	fatores vitais (performance, segurança, etc.) que interferem em projetos ágeis. Os autores aplicaram em 3 estudos de caso pequenos (aproximadamente 30 a 50 story points cada)
<i>Métricas para avaliar a qualidade (acompanhamento do processo, métricas para qualidade do produto etc.)</i>	
Kane (2007)	Sugere unidades de medidas que poderiam ser utilizadas para identificar o tamanho de projetos ágeis (ex. Ideal day, story points, etc.). Apresenta uma tabela definindo o grau de acurácia das estimativas considerando o nível da definição do projeto. O autor não cita nenhuma aplicação das medidas sugeridas.
Johan (2008)	Apresenta uma série de métricas e escalas para mensurar a qualidade do processo ágil, classificadas em três dimensões: <ul style="list-style-type: none"> - Gerenciamento de requisitos (ex. Quantidade de vezes o escopo foi modificado etc.) - Desenvolvimento (ex. quantidade de histórias que foram implementadas no sprint etc.) - Testes (ex. quantidade de testes automatizados etc.). O autor não cita nenhuma aplicação das medidas sugeridas
Sato (2009) Sato (2007)	Apresenta uma série de métricas para mensurar a qualidade do processo ágil, classificadas em duas dimensões: <ul style="list-style-type: none"> - métricas organizacionais (ex. funcionalidade testada e entregue, tempo médio do ciclo etc.) - métricas de acompanhamento (velocidade, fator de teste etc.). Um dos trabalhos do autor aplica as métricas sugeridas em cinco projetos na Universidade de São Paulo (USP) e em dois projetos governamentais na Assembleia Legislativa do Estado de São Paulo (ALESP).
Ikoma et al (2009)	Os autores sugerem métricas para avaliar a agilidade e eficiência do processo. Segundo os autores as métricas podem ser aplicadas a qualquer metodologia. Por exemplo a eficiência é obtida por meio da relação de valor total das entregas dividida pela quantidade média de resultados do período. Os autores aplicaram as métricas sugeridas em uma organização de software, em 7000 projetos nos últimos 9 anos.
Bonfim (2013)	O autor sugere uma série de métricas para desenvolvimento ágil. Por exemplo quantidade de interrupções, valor entregue, quantidade de itens não previstos etc. O autor não apresenta nenhuma aplicação no meio acadêmico ou comercial.

4. METODOLOGIA DA PESQUISA

Considerando os dados obtidos através da revisão de literatura, foi elaborado o modelo conceitual que representa os conceitos adotados e os relacionamentos entre eles. O modelo conceitual construído (Figura 1) baseou-se na constatação de que as métricas para métodos de desenvolvimento ágil podem ser classificadas em métricas de produtividade e qualidade. Assim, as métricas identificadas pela pesquisa possibilitam medir:

- Produtividade: estimativa de tamanho, custo, prazo, complexidade, dinamismo, quantidade de atividades, quantidade de histórias etc.
- Qualidade: quantidade mudanças, qualidade do processo, proporção entre desenvolvedores e testadores, qualidade dos recursos, quantidade defeitos do produto, reusabilidade do produto, qualidade do produto, tempo médio do ciclo, tempo gasto de integração, satisfação da equipe, expectativa do cliente etc.

A identificação, análise e avaliação da aplicabilidade dessas métricas é essencial para que o monitoramento e execução do processo ágil e a entrega do produto possa ser realizada de forma eficaz e com maior qualidade.

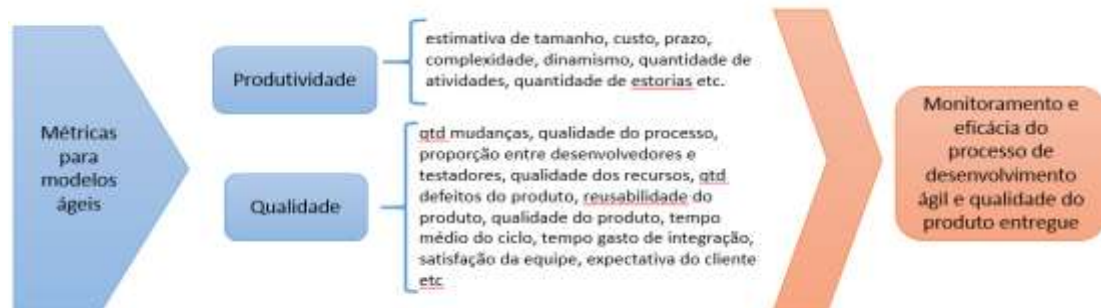


Figura 1: Modelo conceitual.

Assim, essa investigação utilizou a abordagem qualitativa. No que tange ao alcance temporal, a pesquisa se caracterizou como uma pesquisa interseccional, uma vez que investiga os métodos ágeis e as métricas para métodos ágeis de desenvolvimento, em determinado período de 2007 a 2013.

O método adotado foi o estudo de caso e, os instrumentos de coleta de dados que aplicados foram a revisão de literatura e o método Delphi. O estudo de caso foi escolhido pela relação observada entre o objeto e as características básicas do estudo de caso, descritas por Yin (2001): fenômeno observado em seu ambiente natural; estudo exploratório, descritivo ou explanatório uma ou mais entidades (pessoa, grupo, organização) são examinadas, a complexidade da unidade é estudada intensamente, não são utilizados controles experimentais ou manipulações.

O estudo baseado na revisão de literatura abrangeu os métodos ágeis, suas métricas e aplicações. Foram pesquisados a Biblioteca digital de teses e dissertações e o Portal da Capes (somente os artigos acessíveis) no período de agosto/2013 e janeiro/2014. Foram pesquisados os strings: “método ágil”, “métodos ágeis” e ‘métrica’, “mensuração”.

Considerando o caráter exploratório e descritivo da pesquisa e a pouca aplicação (acadêmica e comercial) das métricas identificadas na revisão da literatura optou-se pelo método Delphi com Q-Sort para obter a percepção e consenso de técnicos sobre as métricas mais aplicáveis para o ambiente de desenvolvimento que utilize a metodologia ágil. A realização de uma pesquisa utilizando o Delphi ocorre mediante sucessivos questionamentos a um grupo de especialistas. As respostas são cumulativamente analisadas para analisar a obtenção ou não de consenso. De acordo com Wright e Giovinazzo (2000), o consenso no Método Delphi representa uma consolidação do julgamento intuitivo do grupo participante.

O método Delphi com Q-sort é um método de pesquisa que é utilizado em diversas áreas de conhecimento como técnica de comunicação em grupo particularmente em situações onde se pretende alcançar consenso de opinião (SOARES, 2009). Neste tipo de estudo os peritos são postos perante um conjunto de cartões contendo declarações sobre um determinado assunto. São então solicitados a ordenar estes cartões segundo uma escala contínua de “pouco importante” a “muito importante”, segundo a sua opinião. Encontram-se trabalhos na literatura (SAMARTINHO, 2013; CASTRO ET AL, 2012) em que o Q-sort tem sido utilizado sem os cartões, mas solicitando que os peritos identifiquem em uma lista de declarações a ordem de importância, valor etc.

A Q-Sort na visão de Thomas & Watson (2002) apresenta alguns benefícios, tais como: oferece um meio para um estudo em profundidade para pequenas amostras; pode ajudar a investigação exploratória; captura a subjetividade com a mínima interferência do investigador; os participantes não precisam ser selecionados aleatoriamente; as suas técnicas de análise ajudam a proteger os respondentes da influência do investigador.

Assim, foi elaborada uma lista com 21 métricas de produtividade e uma lista com 20 métricas de qualidade que foram apresentadas aos técnicos/peritos. Para direcionar as questões do Delphi foram identificadas e analisadas as métricas mais citadas e aplicadas no universo acadêmico e/ou comercial, considerando o material pesquisado. O Delphi com Q-Sort foi aplicado em três rodadas. Foi utilizado o software SPSS Statistics para análise dos resultados.

Para pesquisar sobre a aplicabilidade das métricas, foi escolhido o Uniceub – Centro de Ensino Universitário de Brasília. A escolha justificou-se pois é uma empresa de Brasília (escopo da pesquisa), uma empresa de grande visibilidade e, sua área de TI trabalha com métodos ágeis (SCRUM). Na sua área de TI trabalham 22 técnicos e gerente com esse método ágil. Foi definida assim a população da pesquisa como 22 indivíduos.

5. ANÁLISE DOS RESULTADOS

A análise do perfil dos participantes identificou-se que 68% da população possui graduação superior, variando entre o ensino superior completo, lato senso e stricto senso. Cercar de 66% da população pesquisada trabalha com método ágeis há mais de 2 anos. Com relação a percepção do seu grau de conhecimento sobre métodos ágeis 95% classifica entre regular e bom e, destes 72% classifica seu conhecimento em métodos ágeis como bom.

5.1 MÉTRICAS DE PRODUTIVIDADE E QUALIDADE – 1ª. RODADA DO DELPHI COM Q-SORT

As Tabelas com as métricas de produtividade e qualidade ordenadas por ordem alfabética e com a respetiva descrição, foi apresentada aos 22 membros do painel de especialistas. A primeira rodada foi realizada em Maio de 2014. O questionário das métricas de produtividade foi impresso e entregue para preenchimento pelos peritos. Obteve-se uma taxa de resposta 20 membros, isso corresponde a aproximadamente 90%. O questionário das métricas de qualidade foi impresso e entregue para preenchimento pelos peritos. Obteve-se retorno de 21 participantes, ou seja, uma taxa de resposta de aproximadamente 95%. A Análise estatística revelou um valor do coeficiente Kendall's W de 0,402 para as métricas de produtividade e de 0,309 para as métricas de qualidade (Figura 2). O que significa uma concordância média dos membros do painel não sendo estatisticamente significativa, pelo que se promoveu uma segunda rodada.

De acordo com Siegel (1975), o coeficiente de concordância de Kendall é um método não paramétrico que busca verificar qual o grau de associação ou correlação (concordância) entre um conjunto de 3 ou mais variáveis. Como todo método não paramétrico, não exige pré-especificações quanto ao tipo de distribuição da população ou qualquer outro parâmetro. Para utilizar o método, cada variável deve estar pelo menos no nível ordinal, para possibilitar a ordenação dos escores de cada variável em postos. O valor do coeficiente W está compreendido entre 0 e +1, sendo 0 um valor que significa falta de correlação entre as variáveis, enquanto que +1 significa uma correlação perfeita (SIEGEL, 1975).

Com relação ao nível de significância assintótica, tipicamente, um valor menor do que 0,05 é considerado signifiicante. A significância assintótica é baseada no pressuposto que o tamanho da amostra foi suficientemente grande. Ao contrário, se o tamanho da amostra é

pequeno, poder-se-ia obter uma pobre indicação de significância. Nesse caso o nível de significância assintótica é 0, o que demonstra que o tamanho da amostra é significativo.

Test Statistics		Test Statistics	
N	20	N	21
Kendall's W ^a	,402	Kendall's W ^a	,309
Qui-quadrado	160,873	Qui-quadrado	123,147
df	20	df	19
Significância Assintótica	,000	Significância Assintótica	,000
a. Coeficiente de concordância W de Kendall		a. Coeficiente de concordância W de Kendall	

Figura 2 – Testes estatísticos de concordância – Métricas de produtividade e Qualidade – 1ª. Rodada do Delphi com Q-sort

5.2 MÉTRICAS DE PRODUTIVIDADE E QUALIDADE – 2ª. RODADA DO DELPHI COM Q-SORT

Nesta rodada, solicitou-se aos membros do painel que voltassem a ordenar os fatores constantes da lista de métricas de produtividade seguindo os procedimentos do Q-Sort, utilizando-se a consolidação da lista de fatores ordenada na primeira rodada. Esta segunda rodada realizou-se em Maio 2014, tendo participado 22 peritos, o que corresponde a uma taxa de resposta de 100 %. A Análise estatística revelou um valor do coeficiente Kendall's W de 0,675 para as métricas de produtividade e de 0,729 para as métricas de qualidade (Figura 3), o que significa uma concordância alta dos membros do painel e sendo estatisticamente significativa. Mesmo assim, decidiu-se promover uma terceira rodada.

Test Statistics		Test Statistics	
N	22	N	22
Kendall's W ^a	,675	Kendall's W ^a	,729
Qui-quadrado	296,929	Qui-quadrado	304,514
df	20	df	19
Significância Assintótica	,000	Significância Assintótica	,000
a. Coeficiente de concordância W de Kendall		a. Coeficiente de concordância W de Kendall	

Figura 3 – Testes estatísticos de concordância – Métricas de produtividade e Qualidade – 2ª. Rodada do Delphi com Q-sort

5.3 MÉTRICAS DE PRODUTIVIDADE E QUALIDADE – 3ª. RODADA DO DELPHI COM Q-SORT

Nesta terceira rodada, solicitou-se aos membros do painel que voltassem a ordenar os fatores constantes da lista de métricas de produtividade seguindo os procedimentos do Q-Sort, utilizando-se a consolidação da lista de fatores ordenada na segunda rodada. Esta terceira rodada realizou-se em Junho 2014, tendo participado 20 peritos para as métricas de produtividade e qualidade (90%). Os resultados estão apresentados na Tabela 2 (métricas de produtividade) e Tabela 3 (métricas de qualidade). A Análise estatística revelou um valor do coeficiente Kendall's W de 0,850 para as métricas de produtividade e de 0,753 para as métricas de qualidade (Figura 4). O que significa uma concordância alta dos membros do painel e sendo estatisticamente significativa.

Tabela 2: Métricas de produtividade – 3ª. rodada do Delphi com Q-sort

Métricas de Produtividade	Soma	Média	Desvio	Variância
---------------------------	------	-------	--------	-----------

			<i>Padrão</i>	
Estórias (ou funcionalidades) Entregues: representa o número total de Histórias implementadas e aceitas pelo cliente.	39	1,95	2,762055	7,628947
Lista de atividades concluídas: quantitativo total de atividades concluídas	51	2,55	1,877148	3,523684
Estimativas Finais: representa o total de pontos (ou horas, ou “horas ideais”) efetivamente reportadas como gastas para implementar as Histórias da iteração.	87	4,35	2,084403	4,344737
Complexidade do Sprint: identifica a complexidade do Sprint por meio da quantificação dos módulos que ele interage com ou pontos de interface com outros módulos.	102	5,1	2,989455	8,936842
Quantidade de interrupções: Essa métrica sugere que a partir de um indicador de interrupções, o qual fique visível tanto para os membros da equipe, quanto para pessoas externas esclareça esse argumento e promova a conscientização de como as solicitações externas interromperam o trabalho da equipe.	130	6,5	2,373095	5,631579
Produtividade Pontos de estórias Entregues: representa o número total de pontos de estórias (story points) implementados e aceitos pelo cliente.	131	6,55	3,103055	9,628947
Estórias (ou Funcionalidades) Testadas: quantidade de funcionalidades testadas.	132	6,6	2,562893	6,568421
Facetime: representa o tempo gasto de cada desenvolvedor com a pessoa do negócio e com outros desenvolvedores de quem é dependente de trabalho.	167	8,35	2,601113	6,765789
Dinamismo: a quantidade de mudanças (inclusões, alterações e exclusões de requisitos, estórias ou features por Mês ou por iteração.	173	8,65	2,942877	8,660526
Estimativas Originais: representa o total de pontos (ou horas ou horas ideais) originalmente estimado pela equipe para todas as Histórias da iteração.	187	9,35	2,758241	7,607895
Requisitos de usuários: representa o número de requisitos de usuário implementados e aceitos pelo cliente	208	10,4	2,326054	5,410526
Quantidade de pontos ou estórias transferidas para a próxima Sprint: representa o total de pontos ou estórias não concluídos e transferidos para a próxima sprint	221	11,05	3,425523	11,73421
Complexidade Ciclomática do produto: mede a quantidade de logica de decisão num único modulo de software. Num sistema orientado a objetos, um modulo é um método.	223	11,15	2,277464	5,186842
Complexidade dos Métodos Ponderados por Classe: mede a complexidade de uma classe num sistema orientado a objetos.	265	13,25	3,338373	11,14474

Cenários de caso de uso: representa o número total de cenários de caso de uso implementados e aceitos pelo cliente	298	14,9	0,91191	0,831579
Integração: É calculado considerando o total de linhas adicionadas, removidas e alteradas na iteração sobre o total de commits. Visa identificar quanto código é alterado antes de ser integrado no repositório	327	16,35	0,988087	0,976316
Tempo Médio de Ciclo: tempo médio do ciclo do sistema	341	17,05	0,998683	0,997368
Pontos de função entregues: representa o número total de pontos de função implementados e aceitos pelo cliente	362	18,1	1,071153	1,147368
Linhas Alteradas: representa o número total de linhas (não apenas código-fonte) adicionadas, removidas e atualizadas no Repositório de Código Unificado.	374	18,7	1,657519	2,747368
Linhas de Código: representa o número total de linhas de código de produção do sistema, descartando linhas em branco e comentários.	389	19,45	2,139233	4,576316

Test Statistics

N	20
Kendall's W ^a	,850
Qui-quadrado	340,195
df	20
Significância Assintótica	,000

a. Coeficiente de concordância W de Kendall

Test Statistics

N	20
Kendall's W ^a	,753
Qui-quadrado	286,050
df	19
Significância Assintótica	,000

a. Coeficiente de concordância W de Kendall

Figura 4 – Testes estatísticos de concordância – Métricas de produtividade e Qualidade – 1ª. Rodada do Delphi com Q-sort

Tabela 3: Métricas de produtividade – 3ª. rodada do Delphi com Q-sort

<i>Métricas de Qualidade</i>	Soma	Média	Desvio padrão	Variância
Cliente – satisfação produto: mede o nível de satisfação do cliente com o produto desenvolvido.	34	1,7	1,592747	2,536842
Defeitos de resolução: quantidade total de Defeitos encontrados após implantação	67	3,35	2,560325	6,555263
Equipe – Moral: mede o humor e a motivação de cada membro da equipe.	93	4,65	3,407036	11,60789
Quantidade de itens não previstos: medida para a quantidade de trabalho não previsto que é identificada ao decorrer das <i>sprints</i> . Esse indicador auxilia na identificação do que está faltando durante as reuniões de planejamento da <i>sprint</i> .	99	4,95	2,4165	5,839474
Cliente - satisfação prazo: satisfação do cliente com relação ao prazo de entrega	121	6,05	5,07289	25,73421
Defeitos pós sprint: - quantidade total de defeitos	145	7,25	2,769667	7,671053

encontrados após o sprint				
Backlog: refere-se a existência de backlog contendo o escopo do trabalho e junto com o painel define as tarefas e os produtos associados, com seus correspondentes esforços que foram estimados pelo consenso entre os participantes. E também relacionando os riscos.	146	7,3	3,278318	10,74737
Quantidade de alterações por estórias: refere-se a quantidade de alterações de determinada estória em determinado Sprint ou iteração.	181	9,05	2,928535	8,576316
Defeitos pós release: Quantidade total de Defeitos encontrados pós-release (indicador de atraso)	191	9,55	3,425523	11,73421
Nível de qualidade das Estórias: identifica o quantitativo de estórias com características como independentes, com valor para o usuário, com tamanho adequado e testáveis.	191	9,55	2,645254	6,997368
Reusabilidade: identificação dos componentes reutilizáveis do sistema considerando os componentes adicionados a biblioteca.	207	10,35	3,03098	9,186842
Automatização de testes: identifica o percentual de testes automatizados vs não automatizados	234	11,7	3,98814	15,90526
Equipe - Utilização do SCRUM: mede a situação atual e desejável com relação as práticas do SCRUM na equipe	235	11,75	3,226127	10,40789
Valor entregue: Para essa métrica para cada história do usuário deve ser atribuído um número ou um valor financeiro. Ao final de um <i>sprint</i> , a equipe deve somar o valor de todas as histórias prontas e atualizar o indicador.	237	11,85	2,99605	8,976316
Lista de Riscos: identifica a lista de riscos associada a cada estória	296	14,8	1,151658	1,326316
Equipe - Cultura: porcentagem da equipe que prefere trabalhar em um cenário caótico ao invés de um cenário ordenado.	318	15,9	1,586124	2,515789
Linhas de Código de Teste: representa o número total de pontos de teste do sistema. Um ponto de teste é considerado como um passo do cenário de um teste de aceitação automatizado ou como uma linha de código de teste de unidade automatizado, descartando linhas em branco e comentários.	322	16,1	2,989455	8,936842
Fator de teste: considera o número total de linhas de código de teste da iteração vs o número total de linhas de código na produção	347	17,35	3,674593	13,50263
Commits: representa o número total de commits efetuados no Repositório de Código Unificado.	355	17,75	0,850696	0,723684
Configuração: refere-se a quantidade de projetos vs total de projetos que utilizam um processo de configuração.	386	19,3	1,809333	3,273684

Além disso, foi usado o coeficiente de correlação rho de Spearman entre a ordem dos fatores obtida na segunda ronda e a sua ordenação após a terceira ronda, tendo-se encontrado

o valor de 0,943 (Figura 4) para métricas de produtividade e 0,995 (Figura 5) para métricas de qualidade.

Correlações				
Rô de Spearman	ClasRond2		ClasRond2	ClasRond3
		Correlações de coeficiente	1,000	,943**
		Sig. (2 extremidades)	.	,000
		N	21	21
	ClasRond3		ClasRond2	ClasRond3
		Correlações de coeficiente	,943**	1,000
		Sig. (2 extremidades)	,000	.
		N	21	21

** A correlação é significativa no nível 0,01 (2 extremidades).

Figura 4 – Testes estatísticos de correlação – Métricas de produtividade – 2ª. e 3ª. rodada

Perante a pouca variação do coeficiente Kendall's W da segunda para a terceira rodada, o valor do rho de Spearman já próximo do valor 1 para as métricas de produtividade e qualidade e o número de fatores envolvidos (21 e 20, respectivamente), foi considerado que a concordância encontrada entre os membros do painel foi satisfatória e foi terminado Delphi com Q-Sort.

Além disso, a análise das médias, desvios padrões e variâncias obtidos pelas três rodadas do Delphi com Q Sort, considerando as métricas de produtividade e qualidade, permite identificar que a dispersão do conjunto de valores em análise tendeu a diminuir. Ou seja, a classificação dada por cada perito na terceira rodada, com relação a cada métrica, ficou relativamente mais próxima das outras classificações (dos outros peritos), que nas rodadas anteriores. Isso ratifica a concordância satisfatória encontrada.

Correlações				
Rô de Spearman	ClasRond2		ClasRond2	ClasRond3
		Correlações de coeficiente	1,000	,995**
		Sig. (2 extremidades)	.	,000
		N	20	20
	ClasRond3		ClasRond2	ClasRond3
		Correlações de coeficiente	,995**	1,000
		Sig. (2 extremidades)	,000	.
		N	20	20

** A correlação é significativa no nível 0,01 (2 extremidades).

Figura 5 – Testes estatísticos de correlação – Métricas de qualidade – 2ª. e 3ª. rodada

A Figura 6 apresenta a média das métricas de produtividade por rodada. É possível identificar a existência de métricas com pouca ou nenhuma mudança significativa de médias considerando as três rodadas do Delphi com Q-sort: Estimativas originais, Quantidade de pontos ou histórias, Requisitos de usuários e Tempo médio de ciclo. A Figura 6 também permite identificar as métricas de produtividade que obtiveram uma variação de médias mais significativas durante as rodadas do Delphi com Q-sort, tais como: Complexidade ciclomática do produto, Estórias ou funcionalidades entregues, Lista de atividades concluídas.

A Figura 7 apresenta a média das métricas de qualidade por rodada. É possível identificar a existência de métricas com pouca ou nenhuma mudança significativa de médias considerando as três rodadas do Delphi com Q-sort: Backlog, Equipe utilização do SCRUM, Reusabilidade. A Figura 7 também permite identificar as métricas de qualidade que obtiveram uma variação de médias mais significativas durante as rodadas do Delphi com Q-sort, tais como: Cliente satisfação produto, Commits, configuração, equipe cultura, fator teste, Quantidade de itens não previstos.

Os resultados mostram que, no contexto do estudo de caso analisado as métricas Estórias (ou funcionalidades) entregues (média na 3ª. rodada de 1,95), Lista de atividades concluídas (média na 3ª. rodada de 2,55) e estimativas finais (média na 3ª. rodada de 4,35) foram as métricas de produtividade consideradas de maior aplicabilidade no contexto do estudo. Foram consideradas as métricas de produtividade com menor aplicabilidade, no contexto estudado: pontos de função entregues (média na 3ª. rodada de 18,1), linhas alteradas (média na 3ª. rodada de 18,7) e linhas de código (média na 3ª. rodada de 19,45).

Com relação as métricas de qualidade foram identificadas com maior aplicabilidade: Cliente – satisfação do produto (média na 3ª. rodada de 1,7), Defeitos de resolução (média na 3ª. rodada de 3,35) e Equipe- moral (média na 3ª. rodada de 4,65). E como métricas de menor aplicabilidade: Fator de teste (média na 3ª. rodada de 17,35), Commits (media na 3ª. rodada de 17,75) e Configuração (média na 3ª. rodada de 19,3).



Figura 6 – Média por métrica de produtividade e por rodada do Delphi com Q-sort



Figura 7 – Média por métrica de qualidade e por rodada do Delphi com Q-sort

6.CONCLUSÕES

O objetivo geral desta pesquisa foi identificar e analisar as métricas voltadas para os métodos ágeis e contextualizar sua atuação no processo ágil de desenvolvimento. Foram definidos os seguintes objetivos específicos deste trabalho:

- Identificar os métodos ágeis mais citados pela literatura, suas fases e atividades;
- Identificar as métricas propostas de tamanho, esforço e custo para esses métodos;

- Analisar a aplicabilidade das métricas mais citadas no contexto de um desenvolvimento ágil.

Para isso foram conceituados o processo de desenvolvimento ágil e identificadas suas fases e atividades e foram identificados os métodos ágeis mais utilizados. Identificaram-se as métricas propostas de tamanho, esforço e custo para esses métodos e, considerando os resultados foi analisada a aplicabilidade das métricas mais citadas no contexto de um desenvolvimento ágil.

O método adotado foi o estudo de caso e, os instrumentos de coleta de dados que aplicados foram a revisão de literatura e o método Delphi com q-sort. O instrumento elaborado para aplicação do método Delphi com q-sort foi construído considerando a revisão de literatura elaborada e foram classificadas métricas referentes a produtividade e qualidade.

O estudo de caso foi realizado na área de desenvolvimento de software do Uniceub, onde trabalham 22 técnicos e gerente com método ágil. Foram aplicadas 3 rodadas do Delphi com q-sort e a terceira rodada apresentou um nível de concordância satisfatório. Para análise dos dados utilizou-se o software estatístico SPSS, o coeficiente Kendall's W e coeficiente de correlação rho de Spearman.

Os resultados mostram que, no contexto do estudo de caso analisado as métricas Estórias (ou funcionalidades) entregues, Lista de atividades concluídas e estimativas finais foram as métricas de produtividade consideradas de maior aplicabilidade. Foram consideradas as métricas de produtividade com menor aplicabilidade, no contexto estudado: pontos de função entregues, linhas alteradas e linhas de código.

Com relação as métricas de qualidade foram identificadas com maior aplicabilidade: Cliente – satisfação do produto, Defeitos de resolução e Equipe- moral. E como métricas de menor aplicabilidade: Fator de teste, Commits e Configuração.

7.REFERÊNCIAS

- BHALERAO, S. & INGLE, M.** Incorporating vital factors in agile estimation through. International Journal of Computer Science and Applications, Vol. 6, No. 1, p. 85-97, 2009.
- BONFIM, M.** A importância das métricas para equipes ágeis. 2013. Revista Engenharia de software. Disponível em: <<http://www.devmedia.com.br/a-importancia-das-metricas-para-equipes-ageis/28542#ixzz2idwKEJhi>>. Acesso em 21 Out 2013.
- BRASIL.** Lei nº 8.666, de 21 de junho de 1993. Regulamenta o art. 37, inciso XXI, da Constituição Federal, institui normas para licitações e contratos da Administração Pública e dá outras providências. 1993. Disponível em:<http://www.planalto.gov.br/ccivil_03/Leis/L8666cons.htm>. Acesso em: 16 ago. 2011.
- BRASIL.** Instrução Normativa SLTI nº 4, de 19 de maio de 2008. Dispõe sobre o processo de contratação de serviços de Tecnologia da Informação pela Administração Pública Federal direta, autárquica e fundacional. 2008. Disponível em: <<http://www.governoeletronico.gov.br/biblioteca/arquivos/instrucao-normativa-no-04-2>>. Acesso em: 26 fev. 2011.
- CAGLEY, T.** Agile Estimation Using Functional Metrics. White paper. David Consulting Group, 2009.
- CARVALHO, B. V. de & MELLO, C. H. P.** Implementation of scrum agile methodology in software product project in a small technology-based company. Gestão & Produção, v. 19, n. 3, p. 557-573, 2012.
- CASTRO, C. C.; ANDRADE, A. M. de ; LAGARTO, J. .** Identificação de fatores facilitadores da utilização de recursos educativos digitais pelos professores: a perspectiva de especialistas num estudo e-delphi . Revista Iberoamericana de Evaluación Educativa 2012 - Volume 5, Número 3, 2012.
- COHN, M.** Agile Estimating and Planning. Massachusetts: Pearson Education, 2006.
- FADEL, A. C. & SILVEIRA, H. da M.** Metodologias ágeis no contexto de desenvolvimento de software_XP_SCRUM_Lean. Trabalho de conclusão. Pós Graduação da Faculdade Tecnologia da Universidade Estadual de Campinas. 2010.



- FENTON, N.& PFLEEGER, S.** Software metrics a rigorous & practical approach. 2nd. Ed., PWS Publishing Company. 1997.
- GAMBA, M. L.& BARBOSA, A. C. G.** Aplicação de Métricas de Software com Scrum, Universidade do Extremo Sul Catarinense (UNESC), 2009
- IFPUG.** CPM – Counting Practices Manual. v.4.3. IFPU – International Function Points Users Group, 2010
- IKOMA, M.; OOSHIMA, M.; TANIDA, T.; OBA, M.; SAKAI, S.** Using a Validation Model to Measure the Agility of Software Development in a Large Software Development Organization. Proceedings in: ICSE'09, May 16-24, Vancouver, Canada, 2009
- JOHAN, P.** Quantitative Approach for Lightweight Agile Process Assessment. Master's Thesis UNIVERSITY OF TURKU Department of Information Technology, 2008.
- KANE, B.** Estimating and Tracking Agile Projects. PM World Today, (Vol. IX, Issue V), May 2007.
- KANG, S.; CHOI, O.; BAIK, J.** Model-based Dynamic Cost Estimation and Tracking Method for Agile Software Development. 9th IEEE/ACIS International Conference on Computer and Information Science. 2010.
- KNIBERG, H.** Scrum and XP from the Trenches, USA: C4Media, 2007.
- RAM, P.** Agile Metrics: A seminal approach for calculating Metrics in Agile Projects. Disponível em <<http://www.articlesbase.com/project-management-articles/agile-metrics-a-seminal-approach-for-calculating-metrics-in-agile-projects-3130252.html>>. Acesso em <20/01/2013>.
- SILVA, B. S.; CARVALHO, G. R. de; SANTOS, O. A. R. dos.** Adaptação na Prática de um Setor Público às Metodologias Ágeis. Monografia. Ciência da Computação. PUC-RJ, 2012
- SOMMERVILLE, I.** Engenharia de Software. 5ª. Ed. São Paulo: Prentice Hall, 2007.
- RAM, P.** Agile Metrics: A seminal approach for calculating Metrics in Agile Projects. Disponível em <<http://www.articlesbase.com/project-management-articles/agile-metrics-a-seminal-approach-for-calculating-metrics-in-agile-projects-3130252.html>>. Acesso em <20/01/2013>.
- SAMARTINHO, J.** MODoCOMPETE e-LÍDER: Estudo Delphi com Q-Sort sobre a compreensão de competências específicas e características em e-Liderança. In: 1º. Simpósio internacional - 7º EIN - Ciberespaço: liderança, segurança e defesa na sociedade em rede, Evora, Portugal, 2013
- SATO, D. T..** Uso eficaz de métricas em métodos ágeis de desenvolvimento de software. São Paulo, 2007. Dissertação (Mestrado em Ciências) – Instituto de Matemática e Estatística, Universidade de São Paulo.
- SATO, D. T.** Métricas de acompanhamento para metodologias ágeis. 2009. Revista de Engenharia de Software. Disponível em: < <http://www.devmedia.com.br/artigo-engenharia-de-software-12-metricas-de-acompanhamento-para-metodologias-ageis/12562> >. Acesso em 21 Outubro de 2013.
- SCHWABER, K.** Scrum development process. In: Business Object Design and Implementation. Springer London, 1997. p. 117-134.
- SIEGEL, S.** Estatística não-paramétrica. São Paulo: McGraw-Hill. 1975.
- SOARES, D.** Interoperabilidade entre Sistemas de Informação na Administração Pública. Tese de Doutorado. Universidade do Minho, Portugal, Setembro 2009.
- TANIGUCHI, K. & CORREA, F. E.** Metodologias ágeis e a motivação de pessoas em projetos de desenvolvimento de software. Revista de Ciências exatas e tecnologia, Anhuera Educacional, v.4, n.4, 2009.
- THOMAS, D. M.& WATSON, R. T.** Q-Sorting and MIS Research: A Primer, Communications of the Association for Information Systems, 8, (2002), 141-156
- VICENTE, A. A.** Definição e gerenciamento de métricas de teste no contexto de métodos ágeis. Dissertação. Instituto de ciências matemáticas e computação, Universidade São Paulo - USP, 2010.
- WILLIAMS, L..** A Survey of Agile Development Methodologies. 2007. Disponível em: <<http://agile.csc.ncsu.edu/SEMaterials/AgileMethods.pdf>>. Acesso em <08 Nov 2012>.
- WRIGHT, J. T. C.; GIOVINAZZO, R. A.** Delphi: uma ferramenta de apoio ao planejamento prospectivo. Cadernos de Pesquisa em Administração, São Paulo, v. 1, n. 12, p. 54-65, 2000.
- YIN, R. K.** Estudo de caso: Planejamento e métodos. 2ª. Ed., Porto Alegre: Bookman, 2001.